

Global Intelligence の構想 — 地球規模の「心の社会」実現へ向けて —

沼尾正行

東京工業大学 大学院情報理工学研究科 計算工学専攻

Email: numao@cs.titech.ac.jp

概要: 人間とコンピュータが協調して活動する一つの社会システムとして、広域知能 (GI -- Global Intelligence) の構想について述べる。GI 中の人間を支援し、結合するコンピュータ・システムは、GIANT (GI Associating NeTwork) と呼ばれ、動的なネットワークによる推論機構とリンクの重み付けによる学習機構を持つ。それらの構成と試作結果について述べ、GI の例題について考察することにより、心を社会と見なす「心の社会」だけでなく、社会を心と見なすことから、面白いシステムが考えられることを示す。

Global Intelligence — Towards a global society of mind

Masayuki Numao

Department of Computer Science

Graduate School of Information Science and Engineering

Tokyo Institute of Technology

2-12-1, O-okayama, Meguro-ku, Tokyo, 152 Japan

Abstract: We propose a cooperative system of human beings and computers, called *Global Intelligence* (GI). Human workers in GI are supported by a computer system called *GIANT* (GI Associating NeTwork), which consists of an inference mechanism by using a dynamically transforming network, and has a learning mechanism by adjusting a weight on each link in the network. After showing their structures and experiments, we point out that we can construct an interesting model of intelligence not only by analyzing a *Society of Mind*, but also by synthesizing *Mind of a Society*.

1 はじめに

人間の共同作業を考えてみると、ユーザである人間と、その資産であるデータベースおよびプログラムが広域に分散しており、その中で複数の人間が共同して作業を行っている。そこでは、距離に比例した時間的オーバーヘッドが存在し、それを縮める努力が通信の分野でなされ、電信、電話、電子メールなどの技術が開発されてきた。

ところが、距離的な障壁が克服されてみると、電話をかけた受けたりすることや、電子メールを書いたり読んだりするというような、通信作業そのものの手間が無視できなくなってきた。たとえば、官庁や大学の事務組織を考えてみると、地域的に極めて小さな領域、たとえば、一つの建物を占める組織が活動している場合でも、そこを情報が流れ、活用する際の時間的オーバーヘッドはかなり大きい。その手間を最小限にする手段として、ネットワーク上に有機的に構築されたグローバルな知能、すなわち Global Intelligence (GI) [11] を提案する。

従来の人工知能研究では、人間と同じように知的な機械を構築することが目標となっていたのに対し、GI は人間を模倣するシステムではなく、人間自体をもそのなかに取り込み、機械を道具にしながら、活動する一つの社会システムを指す。GI 中の人間を支援し、結合するコンピュータ・システムを、GI Associating NeTwork (GIANT) と呼ぶ。

CSCW は電話を強力にした手段になっており、実時間性が重視され、相手がそこにいるかのように作業可能な通信システムを目指している。それに対し GI では、手紙や電子メールを知的にすることにより、相手に伝える文章を書いたり、読んだりすることなしに、協調作業を可能にするシステムを構築する。その意味ではコーディネイタ [5] と似ているが、コンピュータ側に、プログラミング可能な知識処理能力による柔軟性や学習能力による自己組織性を持たせるといふ点で、はるかに強力なシステムを目指している。

2 GI の例題

GI の行うべきことをはっきりさせるため、例題をいくつか述べよう。

2.1 広域書類システム

官庁と研究費の書類をやりとりする場合を考えよう。研究費の申請時期が近付くと、まず官庁から申請者の手元に申請書のフォームが送られてくる。フォームには七面倒臭い注意書がついているが、普通、申請者は、そんなものは読み飛ばして欄だけを見ながら適当に文字を記入し¹、もよりの事務室に提出する。事務室では記入内容の形式的なチェック、すなわち、「○×大学工学部」と書くべきところを「○×大学工学部計算学科」と書いていないかとか、金額の帳尻が合っているかなどをチェックし、間違いを指摘して、突き返してくる。その関門を通過すると、組織元縮めの事務部が、その研究費独自の縛りに違反していないか入念に確かめ、問題がなければ官庁に提出する。以上の過程を踏むと一か月程度の時間はかかってしまい、さらに官庁で数か月にわたる審査が「内容」について行われて、半年以上した末に、予算がついてしまったりすると、大慌てで対応に追われることになるわけである。

以上を行う GI は、表計算ソフトウェアが分散したようなシステムになる。官庁から組織の事務部を通じて研究者に送られるフォームの各欄には注意書に載せられる程度の制約条件は知識として記述されており、記入時に誤りが指摘される。一人で複数の申請を行うため、申請者は一つのフォームに対して、複数のフォームを返してもよいし、ローカルに試し書きをして注意書の制約を満たすかどうか調べられることもできる。

研究費の募集は無数にある。それがすべての研究者に配布されていたのでは、ネットワークのニュースシステムのようになってしまい、研究者は募集の山に埋もれてしまう。これに対応するため、GIANT は後述するような動的なネットワーク構造で構成され、各リンクにニューラルネットのような重みがついている。学習により、その研究者がよく利用する研究費に通じるリンクの重みは大きく、疎遠なリンクの重みは小さくなり、情報のフィルタリングが行われる。

フォーム添付の注意書を満たした記入内容について、組織の事務部で部局名や番号などの組織独自の制約を満たしているかチェックされる。あらかじめ

¹ 文部省の科学研究費については、ボランティアのグループから L_AT_EX のマクロが提供されるようになったのは、画期的なことである。フォームの枠を生成してくれるだけでなく、欄内容の制約条件まで判定してくれる。

め、予想される制約を知識として記述しておけば、GIANT が自動的に検査するし、申請書が集まるに従い、問題点が出てくれば、動的に知識を加えていけばよい。最終的には人手で検査され、GIANT を通じて、問題点が申請者にフィードバックされ、締切の催促なども設定によって自動的に行われる²。

官庁では、以上の過程で書類の形式上の問題はすでに解決しているから、受け付けた書類は直ちに審査に回すことができる。審査で評点の高い申請を選びだす操作などは自動的に GIANT が行い、最終的な判断は手動で行う。

以上のように、書類のやりとりによる研究費審査の過程を人間と機械の混在下で行う GI が考えられる。

2.2 ソフトウェアの配布とバージョンアップ

インターネットではフリーソフトウェアが流通しているが、ニュースシステムという文書交換のシステムへの投稿、購読、必要に応じたインストールという形態になっており、全くの人手によって維持されている。これは、コンピュータによりシステムが構築されていることを考えると、あまりに迂遠であり、有用なソフトの投稿やバグ報告などの監視役が常時各組織にいないければ、投稿され続けている膨大な量のソフトウェアの最新バージョンを使うことはできない。このような問題を解決する試みはいくつか提案されているが [14]、ここでは GI によるアプローチを考察する。

ソフトウェアの場合は、広域書類システムにおける申請者が作成者に、審査部がユーザにほぼ対応する。ソフトウェア作成者がプログラムを作成し、自分で評価を行う。ある程度のレベルに達すると、ローカルユーザに公開され、特定の用途において、他のソフトウェアより優れていると、徐々にローカルユーザに使用されて、組織内での評価が上昇する。

解決すべき問題を抱えたユーザがプログラムを選ぶ場合、評価の高いソフトウェアの中から用途が一致するものを見つける。そのため、自然言語による説明を用いて検索したり、帰納論理プログラミング [3] の技術により、試験実行しながら、類推 [13] や原始式の変形 [8] により、探し出すなどの手法を用いることになろう。

²締め切りの催促を無視していると、学習によってリンクの重みが小さくなり、あまり催促が来なくなる？

ソフトウェアをインストールするという概念にとらわれていると人手が絡んでくるのを避けるのは難しい。ソフトウェアは作成者の手元に一つだけ存在し、他のサイトは、それを「キャッシング」していると考える。その際、ソフトウェアは作成された環境をひきずりながら移動する。動作環境をすべてコピーしては、空間的オーバーヘッドが大きくなり過ぎるので、遠隔地にある環境を参照可能にする完全な分散処理機構が基本技術として実現されなければならない。環境中の頻繁に利用される一部分だけを再びキャッシングすることで、効率的な実行環境を保証するのである。

研究室での評価が揺るぎないものになると、さらに α 、 β テストユーザを経て、一般社会のユーザにも使われるようになる。そこでも当然、再度評価が行われるし、バグにより評価が下がったり、新バージョンが出現して、その評価が上がってくると、旧バージョンは置き換えられてしまうこともある。新バージョンのバグが深刻な場合には、旧バージョンが使い続けられることもあろう。GIANT はそうしたダイナミックな使用形態に対し、通信経路上のリンクの重みを調整することで対処する。評価の高いバージョンへのリンクの重みを高く調整するわけである。

研究室のような一般的な集団において使われることはなくても、マニアによって使われるということもある。このような階層性を破った社会構造も、当然受容される。

評価過程のかかなりの部分は自動化されるだろうが、最終的には個々の人間の判断結果を蓄積することになる。従来は評価結果は、個人の頭の中にあたり、限られた集団の共有知識として、断片的に文書化されていた。GIANT の学習機能により、その評価結果を一般の人が自動的に利用することができる。評価は、リンクの重みとして学習されるので、客観的な評価にローカルなコミュニティにおける評価と個人の主観がうまく織り込まれた結果となる。口コミによるローカルな評価を取り入れることも容易になるわけである。

2.3 発明支援

ソフトウェアの問題を一般化して、アイデアを共有化するためのシステムを考える。特許の出願件数は年々増加しており、今や膨大で検索もままならない状態であり、特許庁による集中管理は限界に

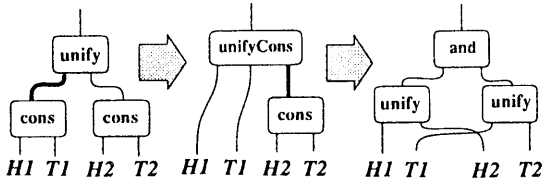


図 1: リスト構造の単一化

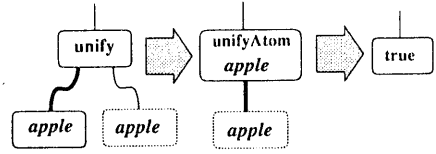


図 2: 定数の単一化

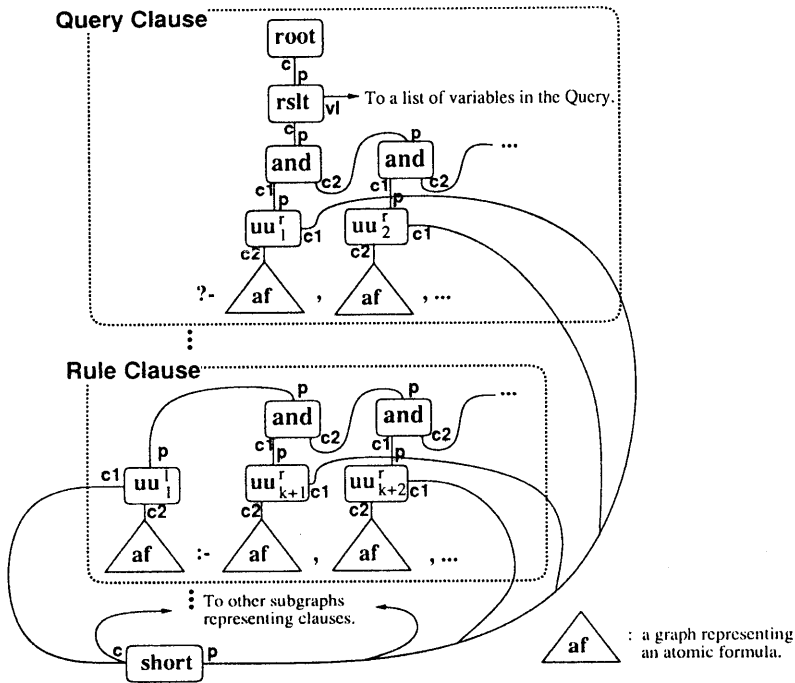


図 3: 初期グラフ

近付きつつある。前節における配布と評価の機能をさらに発展させ、アイデアの分散評価、およびその結果の共有システムを開発することも考えられる。

3 GIANT の機構

GIANT の基礎的機構としては、グラフリダクションによる推論機構 [4, 9] を採用し、上述の例題に必要な調停および学習機能を実現するため、各リンクに重みを付加する [7]。

3.1 動的なネットワークによる分散推論
リスト構造の単一化を行う場合のリダクション過程を図 1 に、定数の場合を図 2 に示す。

リダクション規則は、その適用の可否が図中太線のリンク両端のノード名を調べるだけで判定されるように記述することにし、その記述原則を「ワンアーク・リダクションの原則」と呼ぶ。これに従ったリダクションシステムでは、規則の適用可能性がリンク単位で定まり、競合解消の結果に応じて、リンク単位で簡約化が行えるため、計算機構が単純になり、効率も向上する。また、GIANT では地理的に離れた二点間にリンクがまたがるが、この原則に従えば、両端の間で互いにノード名を交換するだけで、リダクションの可否を決定し、各地点で局所的にグラフを変形できる³。二点間で情報全体をやり

³文献 [12] において、同じような考え方により、特定の機構に依存した手法を提案した。それを一般化した形で述べたのが、

とりするのではなく、簡約化操作により必要なリンクをたどりながら、目的部分の情報に到達するわけである。

変数への代入は、図4に示すように変数から値にリンクを張り替えることでなされる。

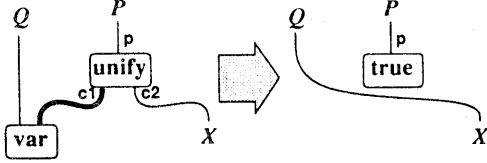


図4: 変数への代入

論理式⁴の全体は、図3に示すような構成のグラフに変換される。各原子式には uu ノードが付けられており、右辺の原子式と左辺の原子式が、short ノードにより結ばれている。

short ノードは図5に示すように両端に結合された uu ノード同士をすべての組合せで結合する。

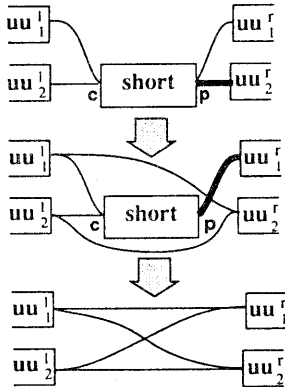


図5: short ノード

こうして結合された uu ノードの組は、図6のように一つの unify ノードに変形されるので、各節の右辺と左辺の原子式同士がすべての組合せで、単一化されることになる。

右辺と左辺の原子式間の単一化は、同じ組合せについては一度しか行わない。すなわち、単一化の過程をあらかじめコンパイルしておくのと等価であり、単一化の繰返しによるオーバーヘッドはない。

図3のグラフでは、右辺と左辺の各原子式間のワンアーク・リダクションの原則である。

⁴図ではホーン節を示してあるが、原理的には一般の節へも拡張可能である。

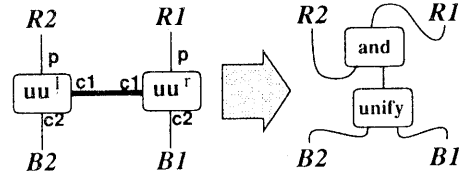


図6: uu ノードの簡約化

単一化が一つの short ノードを介して行われるという意味で、述語定義が中央集権的にアクセスされている。しかしながら、前述した各例題では、局所的な組織が記述された上で、その集まりとして全体の組織が記述されており、述語を1箇所に集めて記述するのは、不可能に近い。この点を解決するには、short ノードを複数にして、それぞれがグループ化された一部の原子式とのみ接続されるようにすればよい。各 short ノードにより、局所的な述語定義が納められた状況 [6] が表わされることになり、広域に広がった組織が記述可能になる。

GI では情報のやりとりが主題なので、述語は状況理論におけるインフォ [2] であると見なした方が自然であろう。情報のやりとりをインフォが飛び交うモデル [16] で表すのである。

3.2 リンクの重み付け機構

ネットワークのリンクには重みが付加されており、情報を得た場合、その評価値が得られるようになっている。リダクションの実行順序は、よい評価値の情報が先に得られるように制御される。研究費の募集、締切の催促やソフトウェアの配布は、この機構により順序づけられ、フィルタリングされる。

GI では、ユーザ個人のシステムの中に評価値が内蔵されるのではなく、ユーザとユーザを結ぶリンクの重みから評価値が計算される。人間社会では、名簿で連絡先を調べて、いきなり飛び込みで面会するのは難しい相手でも、その人に近い人間に紹介してもらった場合には、容易に面会できるのが普通である。そのような紹介によるアクセスは、GIANT において経由する二つのリンクの重みが小さければ可能になるわけである。

情報を受け取ったユーザは、それらが有益であったかどうかの評価を回答する。GIANT は情報を得るのに用いたリンクの重みを変化させることにより、学習を行って、常にユーザの好みを反映した配布を行うようにする。

このような重み付けによる学習機構の実現可能

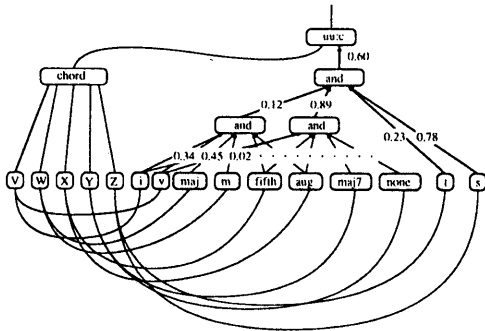


図 7: 重み付けられたネットワーク

性を調査するため、上述のリダクションシステムにおいて、誤差逆伝搬アルゴリズムにより、重みづけを行う実験を行った [7]。対象領域を音楽の自動編曲の学習 [15] とした場合に、学習されたネットワークを図 7 に示す。uu, unify, and ノードは簡約化された結果、成功すると true に簡約化される。これらのノードの出力側、すなわち、親側のリンクを評価リンクと呼び、重み付けは評価リンクにつけられる。図では右半分には評価リンクがあり、重み付けがなされている。リストを構成したり、パターンマッチを行うためのリンクなど、その他のリンクには評価値はつけない。図では左半分にそのようなリンクがある。

ニューラルネットと記号処理を統合する提案は数多くなされているが [1, 10]、グラフィダクションを用いることによる利点は次の 2 点である。

- 命題論理ではなく、述語論理レベルの記述が扱える。
- リダクションにより、ネットワークが簡約化されるので、誤差の逆伝搬が容易になる。

書類やソフトウェアなど構造を持った対象を扱うためには、述語論理レベルの記述が必須である。論理的な推論は多段階に渡ることが普通だが、誤差の逆伝播ではせいぜい数段階程度の修正しか行えず、事実上学習が不可能であった。ところが、再帰を含まない論理式の場合には、リダクションの結果、ネットワークの層の深さは 3 段以下に簡約化され、誤差の逆伝搬は容易になる。再帰を含む場合についても入れ子の数だけ、層の深さが増えるだけである。

手続き型言語や論理プログラムには、実行順序の概念があって、それをハックすることにより、非単調性や副作用など、論理的には取扱いが困難な対

象をうまく組み込んでいる。GIANT では論理的取扱いの難しい対象をリンクの重み付けによりカバーすることになる。

4 おわりに

GI の構想を述べた。心を社会と見なす [17] だけでなく、社会を心と見なすことから、面白いシステムが考えられるのではないだろうか。

参考文献

- [1] ハイライト技術 — ニューラルネットワークと AI の統合 —. (財) 日本情報処理開発協会 (編), AI 白書 — 人工知能の技術と利用, pp. 185-296. コンピュータ・エージ社, 1993.
- [2] K. Devlin. *Logic and information*. Cambridge University Press, Cambridge, 1991.
- [3] S. Muggleton, editor. *Inductive Logic Programming*. Academic Press, London, 1992.
- [4] M. Numao, S. Morita, and K. Karaki. A learning mechanism of logic programs by dynamically sharing substructures. Technical Report 95TR-0001, Department of Computer Science, Tokyo Institute of Technology, 1995.
- [5] ウィノグラード, フローレス (平賀訳). コンピュータと認知を理解する. 産業図書, 1989.
- [6] バーワイス, ベリー (土屋, 鈴木, 白井, 片桐, 向井訳). 状況と態度. 産業図書, 1992.
- [7] 丸谷健介, 沼尾正行. 動的なネットワークを用いた述語論理表現の学習法. 人工知能学会全国大会 (第 9 回) 論文集, 1995.
- [8] 有馬淳. 論理プログラムの自動検索・統合・発見. 情報処理学会研究報告, Vol. 95-AI-99, pp. 165-174, 1995.
- [9] 沼尾正行. 並列グラフィダクションシステムの学習能力 — ネットワーク構造による記号処理と学習. 情報処理学会研究報告, Vol. 93-AI-89, pp. 65-73, 1993.
- [10] 沼尾正行. 複数の情報媒体を用いた学習 — 多戦略学習とその情報源による分析 —. 人工知能学会誌, Vol. 9, No. 6, pp. 837-842, 1994.
- [11] 沼尾正行. Global intelligence の構想. WAL95 : Workshop on Artificial Intelligence and Learning, 1995.
- [12] 沼尾正行, 志村正道. 関数型記号処理言語の分散システムに適した評価法. 情報処理学会論文誌, Vol. 26, No. 2, pp. 247-256, 1985.
- [13] 沼尾正行, 志村正道. メタインタプリタによる帰納的プログラム合成規則の学習. 人工知能学会誌, Vol. 6, No. 6, pp. 920-927, 1991.
- [14] 宇山政志. ソフトウェア製品の流通と個人タスクの統合. 超編集・超流通・超管理のアーキテクチャシンポジウム, pp. 83-92, 1994.
- [15] 沼尾正行. 学習機構による感性情報へのアプローチ — 音楽を題材とした感性の獲得とそれによる感性情報の生成 —. 文部省重点領域研究「感性情報」成果報告書, 1995.
- [16] 中島秀之. 状況推論と会話. 日本認知科学会, Vol. SIGLAI/SIGR&I-91, No. 1, pp. 21-28, 1991.
- [17] ミンスキー (安西訳). 心の社会. 産業図書, 1990.